LEM - An approach for physically based soft tissue simulation suitable for haptic interaction *

Remis BALANIUK
Center for Advanced Techn. in Surgery
Stanford University
remis@robotics.stanford.edu

Ivan F. COSTA
SHARP/GRAVIR
INRIA Rhône-Alpes
Ivan-Ferreira.Costa@inrialpes.fr

Abstract

This paper presents LEM - Long Elements Method, a new approach for physically based simulation of deformable objects, suitable for real time animation and haptic interaction. The method implements a static solution for elastic global deformations of objects filled with fluid based on the Pascal's principle and volume conservation. The physics of the objects are modeled using bulk variables: pressure, density, volume and stress. The volumes are discretised in long elements. This discretisation has two main advantages: the number of elements used to fill an object is one order of magnitude less than in a discretisation based on tetrahedric or cubic elements; the graphic and the haptic feedback can be directly derived from the elements, and no intermediate geometric representation is needed. The use of static instead of PDE equations avoids all the problems concerning numerical integration, ensuring stability for the simulation and for the haptic rendering.

1 Introduction

The method proposed in this paper was conceived for soft tissue real time simulation, particularly for surgical simulation. The priorities in this kind of application are: unrestricted multi-modal interactiveness, including interactive topological changes (cutting, suturing, removing material, etc), physically based behavior, volumetric modeling (homogeneous and non-homogeneous materials) and scalability (high accuracy when needed).

The approach is based on a static solution for elastic deformations of objects filled with uncompressible fluid, which is a good approximation for biological tissues. The volumes are discretised in a set of Long Elements (LE), and an equilibrium equation is defined for each element using bulk variables. The set of static equations, plus the Pascal's principle and the volume conservation, are used to define a system that is solved to find the object deformations and forces. Global and physically consistent deformations are obtained (Fig. 1).

Figure 1: Soft-tissue touched by a rigid probe

For a survey of deformable modeling in computer graphics the reader is referred to [1]. Others recent methods proposed are the "Geometric Nonlinear finite element method" [3], the "Boundary Element Method" [2] and some medical simulators [4], [5], [6].

*Permanent address: (both authors) Universidade Católica de Brasilia - Brazil EPCT Q.S. 7, lote 1 - Águas Claras - 72030-170 - Taguatinga - DF - Brazil

2 Method Formulation

2.1 Pressure and Stress

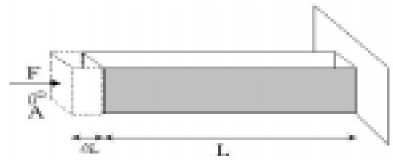

Figure 2: Long element

Consider the long elastic element illustrated in figure 2. The force F per unit of area A is defined as pressure: $P = F/A$. However the force per area unit producing the deformation is also the stress. For small applied forces, the stress s in a material is usually linearly related to its deformation (its change in length in our long elastic object). Defining elasticity E as the variable relating stress and the fractional change in length: $\Delta L/L$, it is possible to write: $s = E\Delta L/L$. Since the stress is related to the fractional change in length, the force can be related to the elongation ΔL in the well known form:

$$F = K\Delta L \qquad \text{where} \qquad K = AE/L. \tag{1}$$

Note that K is not constant, but it depends on the length L.

2.2 Static Solution

The static condition states that the forces, or pressures, in one sense have a correspondent of the same magnitude in the contrary sense on each point of the surface of the object , or: $P_{int} = P_{ext}$. The external pressure P_{ext} on the surface is affected by the atmospheric pressure and by the stress when an elongation exists, so:

$$P_{ext} = P_{atm} + E\Delta L/L. \tag{2}$$

The surface tension also affects the external pressure, as described further in section 2.4.

Considering that the object is filled by fluid, the internal pressure (P_{int}) is formed by the pressure of the fluid (without gravity) and the effect of the gravity acceleration (g), so:

$$P_{int} = P_{fluid} + dgh \tag{3}$$

where h is the distance between the upper part of the fluid and the point where the pressure is calculated. From the last three equations, a continuous equation can be obtained as:

$$E\Delta L/L - \Delta P = dgh \tag{4}$$

where $\Delta P = P_{fluid} - P_{atm}$.

Another external pressure to be considered comes from contacts between the object and its environment. At the points on the object surface, where are some external contacts, a term is added to the right side of equation 2. To obey the action-reaction law, the force applied to the external contact and to the object must to have the same magnitude. It means that the external pressure applied by the contact must be equal to ΔP. The elongation ΔL is defined by the penetration of the contact in order to make the surface follow the contact position (y). With these considerations, the equation 4 can be rewritten for the elements where there is external contact as:

$$\Delta L = y. \tag{5}$$

2.3 Long Elements

To simulate a deformable object we propose a discretisation of its volume in a set of long elements (Fig. 2). The idea is to fill the volume with long elements, to define equilibrium equations for each element based on the stated principles and to add global constraints in order to obtain a global physical behavior. A long element can be compared to a spring fixed in one extremity and having the other extremity attached to a point in the movable object surface. Different meshing strategies can be conceived to fill the objects. Applying the continuous equations (eqs. 4 and 5) for each of this elements we obtain:

$$E_i.\Delta L_i/L_i - \Delta P_i = d_i g_i.h_i \tag{6}$$

for the untouched elements. For the touched elements we obtain:

$$\Delta L_i = y_i \tag{7}$$

To make the connection between the elements two border conditions are applied:

1. Pascal's principle says that *an external pressure applied to a fluid confined within a closed container is transmitted undiminished throughout the entire fluid*. Mathematically:

$$\Delta P_i = \Delta P_j \text{ for any } i \text{ and } j. \tag{8}$$

   The first equation of this section (eq. 6) can then be written without the index i in the term ΔP_i.

2. The fluid is considered incompressible. It means that the volume conservation must be guaranteed when there is some external contact to the object. The volume dislocated by the contact will cause the dislocation of the entire surface, or in other words, the variation of volume due to the elements touched by the contact have to be equal to the sum of the volume created by the dislocation of all untouched elements to ensure the volume conservation:

$$\sum_{i=1}^{N} A_i \Delta L_i = 0 \tag{9}$$

   where N is the total number of elements.

2.4 Surface Tension

To reproduce the surface tension forces a number of terms will be added to the right side of the equation 2 corresponding to the neighborhood considered around the element. These terms are of the form $P = FA = kxA$, where x is the difference between the deformations of an element and its neighbor and k is a local spring constant. For a given element i the term relating its deformation to the deformation of its neighbor j is:

$$k_j(\Delta L_i - \Delta L_j)A_i \tag{10}$$

3 Mathematical Solution

Equations 6, 8 and 10 define the final equation for the untouched elements (considering 4 neighbors):

$$(E_i/L_i + 4kA)\Delta L_i - kA(\Delta L_{i-1} + \Delta L_{i+1} + \Delta L_j + \Delta L_l) - \Delta P = d_i g_i . h_i \tag{11}$$

where k and A were done constant for all elements to make easier the notation.

The untouched elements (equation 11) plus the elements in contact with the environment (equation 5) define a set of N equations, where N is the number of elements used to fill the object. Adding the equation of volume conservation (eq. 9) we have $N+1$ equations and $N+1$ unknowns: the pressure (ΔP) and the deformation of each element (ΔL_i for $i = 1$ to N).These $N+1$ equations can be written as a problem of the type $A.x = B$.

4 Method Implementation

The described method was used to implement a generic soft tissue VR simulator. The simulator was implemented in C++ in a Windows NT platform. This first prototype simulates deformations of a compliant object contacted by a rigid probe.

4.1 System organization

The system is organized around three decoupled main loops, executed concurrently in different processing units (threads, process and/or machines). The first loop simulates the deformations, the second renders the graphics and the third renders the haptics. The main loops share the data structure containing the long elements.

The objects are discretised in Cartesian meshes, each mesh containing long elements parallel to one axis of the reference frame. A Cartesian mesh defines a grid of parallel elements crossing the object. Each element starts in a point of the surface and crosses the volume until the end of the material, defining a line segment parallel to one of the reference frame axis.

Simulation loop The iterative biconjugate gradient method [7] is used to solve the system of equations defined in section 3. The static equations system does not demand any particular concern about time steps, stiffness or stability. The matrix is dynamically defined and the system $A.x = B$ can be rapidly solved. The solution (x) is the surface deformation, defined by a set of length differences in each element, and the difference in pressure.

Graphic loop OpenGL and GLUT are used to render the 3D volumes. There is no explicit geometric model of the object surface. In order to draw the object we use vertices directly derived from the long elements extremities and polygons defined between neighbor elements.

Haptic loop The LE representation of a volume is excellent for haptic rendering. The one point collision detection between the haptic probe position and the volume can be easily done using directly the LE cartesian meshes. Each mesh defines a grid, or a space filling map, and the collision detection in one mesh consists in checking the grid position corresponding to the probe position to see if the probe is penetrating a LE. Each mesh being parallel to one axis of the reference frame, the force feedback estimation is naturally decomposed. Each component of the force vector is independently estimated using the corresponding LE mesh.

During the collision two forces are been applied by the object to the haptic probe: a force applied by the touched elements (eq. 1) on the direction of the element and a force applied by the fluid inside the object (eq. 12). Multiplying both sides of equation 4 by the contact area A_c and comparing to equations 1 we obtain:

$$F = \Delta P.A_c + dgh.A_c \tag{12}$$

This force is perpendicular to the object surface and depends on the internal pressure, not on the penetration.

4.2 Results

In a standard dual 700MHz PC one iteration of the simulation loop takes about 0.05 seconds for a 600 elements mesh. The haptic interface was implemented using a PHANTOM haptic device (http://www.sensable.com). See figs. 1 and 3 for some examples of deformation. The global deformations are physically consistent and important phenomena such as the movement of all parts of the solid due to the preservation of volume are automatically produced.

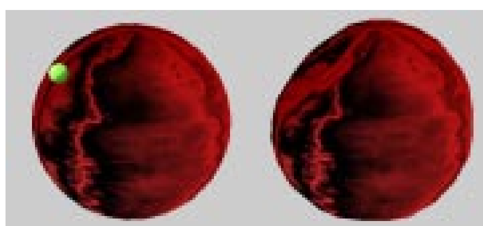

Figure 3: Soft sphere touched by a rigid probe

5 Conclusions

Utilizing the LE method have been able to physically model elastic deformations in a way that preserves volume, permits real time topology changes and is rapidly computable. The discretisation adopted by the method has two main advantages: the number of elements used to fill an object is one order of magnitude less than in a discretisation based on tetrahedric or cubic elements; the graphic and the haptic feedback can be directly derived from the elements, and no intermediate geometric representation is needed. The use of static instead of PDE equations avoids all the problems concerning numerical integration, ensuring stability for the simulation. No pre-calculations or condensations are used, in order to enable real time topology changes.

6 Acknowledgments

The authors would like to acknowledge the Cisco Corporation and SunMicrosystems for their support of this research. We would also like to thank Dr. Kenneth Salisbury, Dr. Thomas Krummel and the CATSS Laboratory at the Stanford University Medical School Department of Surgery; Christian Laugier and the INRIA; and the Catholic University of Brasilia for providing support, encouragment, and facilities.

References

[1] Saraj F. F. Gibson and Brian Mirtich. “A survey of deformable models in computer graphics.” *Technical Report TR-97-19*, Mitsubishi Electric Research Laboratories, Cambridge, MA, November 1997 (http://www.merl.com/reports/TR97-19/index.html).

[2] D. James and D. Pai, ”Artdefo Accurate Real Time Deformable Objects”, *Computer Graphics*, vol. 33, pp: 65–72, 1999.

[3] Yan Zhuang and John Canny. “Haptic Interaction with Global Deformations”. *Proceedings of the IEEE International Conference on Robotics and Automation - ICRA 2000*, 2428-2433, 2000.

[4] Murat Cenk Cavusoglu, Frank Tendick, Micahel Cohn and S. Shankar Sastry. “A Lapraroscopic Telesurgical Workstation”. *IEEE Transactions on Robotics and Automation*, 15(4), 728-739, 1999.

[5] V. Vuskovic, M. Kauer, G. Szekely and M. Reidy. “Realistic Force Feedback for virtual Reality Based Diagnostic Surgery Simulators”, *Proceedings of the IEEE International Conference on Robotics and Automation - ICRA 2000*, 1592-1598 (2000).

[6] Stephane Cotin, Herve Delingette and Nicholas Ayache. “Real-time Elastic Deformations of Soft Tissues for Surgery Simulation”. *IEEE Transaction on Visualization an Computer Graphics*, 5(1), 62-73, 1999.

[7] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling. *Numerical Recipes in C.* Cambridge Univ. Press, 2 ed., 1992.